



## **Integration and Analysis of EPAS and Chassis System in FMI-based co-simulation**

Downloaded from: <https://research.chalmers.se>, 2023-05-05 01:00 UTC

Citation for the original published paper (version of record):

Chen, W., Ran, S., Jacobson, B. (2019). Integration and Analysis of EPAS and Chassis System in FMI-based co-simulation. Proceedings of the 13th International Modelica Conference(157): 717-724.  
<http://dx.doi.org/10.3384/ecp19157717>

N.B. When citing this work, cite the original published paper.

# Integration and Analysis of EPAS and Chassis System in FMI-based Co-Simulation

Weitao Chen<sup>1,2</sup> Shenhai Ran<sup>1</sup> Bengt Jacobson<sup>2</sup>

<sup>1</sup>Vehicle Dynamics CAE, Volvo Cars, Sweden,

{weitao.chen, shenhai.ran}@volvocars.com

<sup>2</sup>Vehicle Dynamics, VEAS, Chalmers University of Technology, Sweden,

bengt.jacobson@chalmers.se

## Abstract

The vehicle steering characteristics and active functions can be virtually developed with a high-fidelity electric power assisted steering (EPAS) model and a multibody chassis model. The simulation of the EPAS model requires small integration step due to high stiffness and interfacing with the controller. The multibody chassis model is computationally heavy for each integration step due to calculation of large matrices. A mono-simulation based on a single solver is not efficient for this case. Instead a co-simulation (solver coupling) approach has been used to overcome the drawbacks.

In this paper the EPAS system and chassis system are modeled in Dymola and further exported as separate functional mockup units (FMUs) and integrated with the control algorithms in Matlab. A co-simulation based on the explicit parallel calculation scheme (Jacobi scheme) has been used. A huge simulation speed-up has shown the potential and effectiveness of the approach. To understand its accuracy and tolerance, analysis on the numerical error and dynamics of the coupled-system are given.

**Keywords:** EPAS system, Chassis system, Co-Simulation, FMU

## 1 Introduction

Modern vehicles involve more electric and functional subsystems with a trend of electrification and automation. Multi-domain subsystems need to be modeled and integrated by co-simulation for a holistic development. This modular approach is quite common because the models might be from multiple sources (e.g., OEM-suppliers relationship) in different tools. Furthermore, it enables each model efficiently solved by a domain-specific numerical method. The approach has been applied in many engineering cases such as an integration of large-scale pantograph-catenary system (Arnold, 2010), a distributed simulation of a 4 cylinder engine (Saidi et al., 2016).

For accurate simulation of vehicle handling, steering and active function tests, a mechanical multibody chassis model and an EPAS model are needed. The chassis model has hundreds of degrees of freedom and its dynamics is relatively slow especially for handling and steering simu-

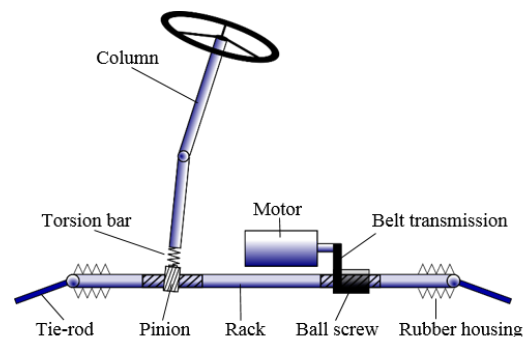
lation on the flat road. The EPAS model has much faster dynamics because of the lightweight components, friction elements, electric parts and the control algorithms. Its fidelity is critical for the steering feel. As the chassis model and EPAS model differ in terms of dynamics and requirements. A mono-simulation based on a single solver might not be the optimal solution. In this paper, a FMI-based co-simulation has been tested. The coupled-system is constituted by FMUs of a chassis model, an EPAS mechanical model, a S-function for the EPAS electrics and control algorithms.

The modeling work in Dymola is presented in Section 2 and Section 3. The integration based on FMI standard and the co-simulation setup are shown in Section 4. In Section 5 the co-simulation results and analysis on simulation speed and system dynamics are discussed.

## 2 EPAS System

### 2.1 EPAS mechanism

The EPAS mechanism comprises mainly a steering wheel and column, a steering rack and an EPAS motor as shown in Figure 1. The steering column is connected to the rack and pinion by a compliant torsion bar. A belt transmission connects the motor and the ball screw which transfers the motor rotation into the rack translation.



**Figure 1.** The EPAS system with axle-parallel drive.

The steering rack is articulated to the vehicle chassis through the suspension tie-rods and steers the front wheels. This mechanical chain builds a direct interaction

between the driver and the road. The auxiliary electric motor can deliver an assist torque according to the torsion bar deflection and vehicle speed to reduce the steering effort. The steering feel defined by the introduced mechanisms is a key metric in vehicle development and needs to be accurately simulated and evaluated on a driving simulator.

The EPAS mechanism is modeled by 3 degrees of freedom: 2 degrees of rotation for the column and motor, 1 degree of translation for the rack. Different from the Modelon Vehicle Dynamics Library (VDL) steering template, the non-linear effect from the column CV-joint has been neglected and no multibody components have been used for simplicity. Instead, the friction and motor dynamics are very important for EPAS in terms of vehicle steering response, subjective feeling for the driver and the stability of EPAS controller (Harrer and Pfeffer, 2017), more detailed effects have been considered.

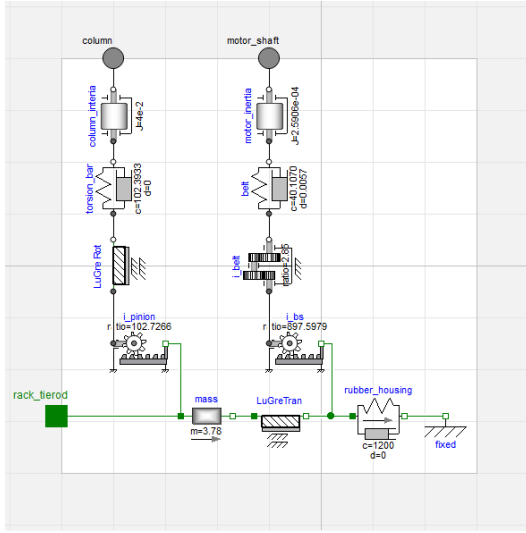


Figure 2. The EPAS mechanism modeled in Dymola.

The model based on basic Modelica mechanical components and detailed friction elements, shown in Figure 2, is created according to the dynamics on the column, the motor and the rack:

$$J_{column} \ddot{\delta}_s = T_s - T_{pinion} - T_{cfriction} \quad (1)$$

$$J_{motor} \ddot{\delta}_m = T_{motor} - T_{belt} \quad (2)$$

$$m_{rack} \ddot{x}_R = F_{pinion} + F_{assist} - F_{rod} - F_{rfriction} - F_{housing} \quad (3)$$

The states and parameters of the model are given in Table 1. The force  $F_{pinion}$  and  $F_{assist}$  are calculated from the respective torques and transmission ratios:

$$F_{pinion} = T_{pinion} / i_{pinion} \quad (4)$$

$$F_{assist} = T_{belt} / (i_{belt} i_{bs}) \quad (5)$$

where the torque  $T_{pinion}$  and  $T_{belt}$  are calculated based on the deflection of the torsion bar and belt.

Table 1. States and parameters of the EPAS model.

Notation	Definition
$\delta_s, \delta_m$	angle of the steering wheel, motor
$x_R$	rack displacement
$J_{column}, J_{motor}$	inertia of the column and wheel, motor
$m_{rack}$	rack mass
$i_{pinion}, i_{belt}, i_{bs}$	transmission ratios of the rack pinion, the belt, the ball screw
$T_s$	steering torque
$T_{pinion}$	torsion bar torque
$T_{cfriction}$	column friction torque
$T_{motor}$	applied torque from the motor
$T_{belt}$	load torque on the output shaft
$F_{pinion}$	force transmitted by the rack pinion
$F_{assist}$	assist force from the ball screw
$F_{rod}$	tie-rod force along the rack
$F_{rfriction}$	friction on the rack
$F_{housing}$	a spring-damper force from the housing

## 2.2 Friction elements

The mechanical friction is mainly divided into the upstream element  $T_{cfriction}$  and the downstream element  $F_{rfriction}$ . The friction elements are modeled by the LuGre friction model (Astrom and Canudas de Wit, 2008). In Modelica the standard friction element is implemented by discrete events switching between stuck and slide mode. An appropriate numerical method is needed for this continuous/discrete approach. The LuGre friction model adds the hysteresis effect and it expresses the friction by differential equations:

$$\dot{z} = v - \sigma_0 z / g(v) |v| \quad (6)$$

$$g(v) = F_c + (F_s - F_c) e^{-(v/v_s)^2} \quad (7)$$

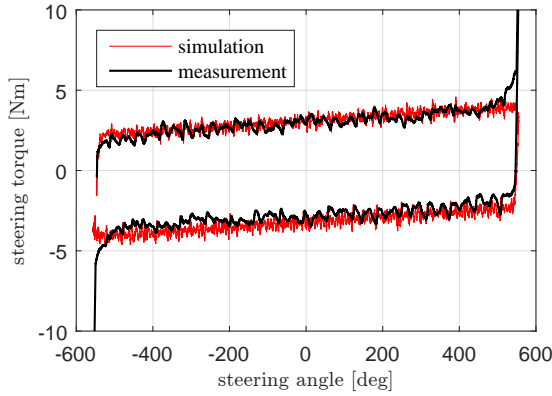
$$F_{friction} = \sigma_0 z + \sigma_1 \dot{z} + \sigma_2 v \quad (8)$$

where  $v$  is the sliding velocity,  $z$  is the internal state. The bristle stiffness  $\sigma_0$  and micro-damping  $\sigma_1$  produce a spring-like behavior in small displacements.  $\sigma_2$  is the viscous friction coefficient.  $g(v)$  is a velocity-dependent term relating to the Coulomb friction  $F_c$ , the static friction  $F_s$  and the Stribeck velocity  $v_s$ .

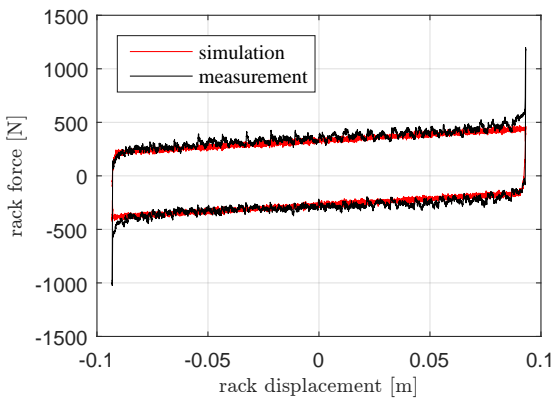
Numerical methods for continuous system can be used to solve this model. However, its dynamics is so stiff that small tolerance value for variable step solver or small time steps for fixed step solver is needed. As a result, the simulation speed gets slow. A detailed implementation and analysis of the LuGre friction model in Modelica has been introduced in (Aberger and Otter, 2002).

The friction model parameters have been partially identified from experiments using a steering system test rig with a steering robot connecting the steering wheel and two linear actuators connecting the rack. Pull-by-torque and pull-by-rack tests (Harrer and Pfeffer, 2017) have been taken with the EPAS controller deactivated. The

steering system is excited accordingly either by velocity-controlled steering wheel input or rack input in free load condition. Thanks to the acausal modeling, the recorded data can be conveniently taken as input to the EPAS model. The comparison of the simulation results and the measurement data are given in Figure 3 and Figure 4.



**Figure 3.** Steering torque in a pull-by-torque test with a steady steering velocity input of 13 deg/s.

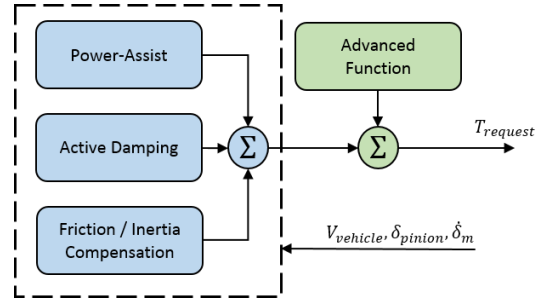


**Figure 4.** Rack force in a pull-by-rack test with a steady rack speed of 2 mm/s.

### 2.3 EPAS control

The large inertia, high friction and less damped behavior from the EPAS mechanism is counterbalanced by the basic steering functions involving the inertia compensation, friction compensation, active damping and power-assist. The advanced driving functions like the lane keeping aid (LKA) and Pilot-Assist are added to the motor torque request  $T_{request}$  in Figure 5, which is further delivered to the electric motor.

The detailed models of the control algorithm, ECU and electrics are provided from the supplier as black-box S-functions with inputs of vehicle speed  $V_{vehicle}$ , torsion bar angle  $\delta_{pinion}$ , motor speed  $\dot{\delta}_m$  and the external request from the advanced functions. So that the system needs to run in the Simulink environment with a forward Euler method with 1 ms integration step.

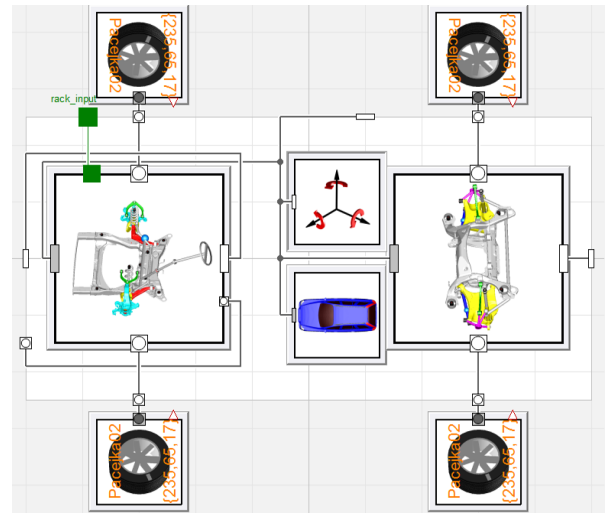


**Figure 5.** A block diagram of the EPAS control architecture.

## 3 Chassis Model in Dymola

A chassis model based on the Modelon VDL has been used. It is constituted by the car body, Pacejka tire models and suspensions (Figure 6). To facilitate the computation, the suspension linkages are represented by kinematic tables. The wheel orientation and translation varies according to the wheel jounce and steering input. A validated model of Volvo XC90 has been used in the work.

To integrate the chassis model with the created EPAS model, 1D translation interface is attached to the rack and the original VDL steering model is disconnected as a dummy part. In this way the translation is still relative to the front subframe whose compliance may have a great impact on the steering feel.

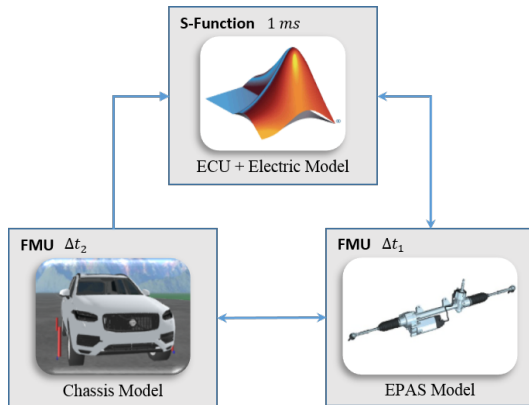


**Figure 6.** The multibody chassis model in Dymola.

## 4 Co-Simulation Setup

The EPAS model and chassis model are compiled to separate FMUs embedded with variable step and variable order Dassl solvers. The EPAS FMU, chassis FMU and EPAS control S-function are coupled by specified input-output signals (Figure 7). At the coupling interface the chassis FMU takes the rack velocity  $\dot{x}_R$  as input and EPAS model takes the force  $F_{rod}$  as input. The decision is based on our analysis from a previous work (Chen et al., 2018), briefly:

- The force variable should be applied towards the heavier and stiffer part for robustness. The EPAS system due to the gear ratio effect is much heavier and stiffer than the lateral dynamics of chassis system.
- The displacement input results an improper dynamic system (more zeros than poles in the transfer function). Thus, the derivation of input variable is needed and this might generate noisy or incorrect results.



**Figure 7.** The layout of the co-simulation setup.

The chassis FMU is setup with a communicative step of  $\Delta t_2$  which is the time size that the local solver updates the input and output. The communicative step of EPAS FMU is  $\Delta t_1$  with a default value of 1 ms because of the coupling with the controller.

For simplicity, the co-simulation is implemented with an explicit parallel calculation scheme (i.e., during the communicative interval each model is integrated independently and the input is approximated from extrapolation). In this work, a common constant extrapolation (zero-order hold) has been used. Although this calculation scheme suffers from the numerical stability and coupling errors. It is advantageous for less computational burden and easy implementation in practice because no control of computation sequence or iterative process is needed from a master algorithm (Busch, 2012).

## 5 Co-Simulation Results

The co-simulation have been tested with various scenarios as given in Table 2. For comparison, a mono-simulation reference, denoted by *Ref-1*, is made by compiling the EPAS and chassis model together as a whole FMU with the same solver. The tests are performed on a laptop with 32GB RAM and one Intel Core i7 processor which runs 8 cores at 2.70 GHz.

### 5.1 Simulation speed-up

A 5 seconds steering maneuver with a sine wave steering torque input has been simulated. From the CPU time of each simulation case (Figure 8), one can see that comparing with *Ref-1* the co-simulation cases are much faster

**Table 2.** Simulation Cases

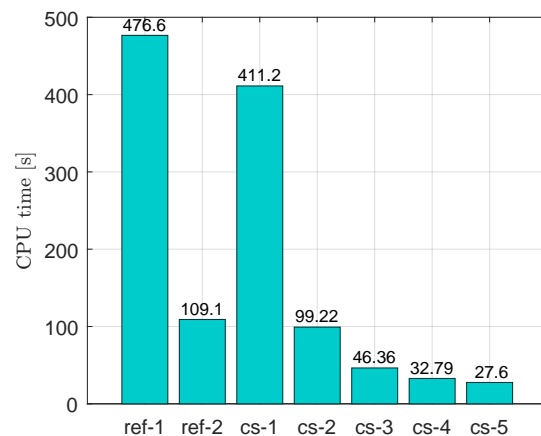
Case	Communicative step	
<i>Ref-1</i>	$\Delta t_1 = 1ms$	no $\Delta t_2$
<i>Ref-2</i>	$\Delta t_1 = 5ms$	no $\Delta t_2$
<i>CS-1</i>	$\Delta t_1 = 1ms$	$\Delta t_2 = 1ms$
<i>CS-2</i>	$\Delta t_1 = 1ms$	$\Delta t_2 = 5ms$
<i>CS-3</i>	$\Delta t_1 = 1ms$	$\Delta t_2 = 10ms$
<i>CS-4</i>	$\Delta t_1 = 1ms$	$\Delta t_2 = 15ms$
<i>CS-5</i>	$\Delta t_1 = 1ms$	$\Delta t_2 = 20ms$

especially when  $\Delta t_2$  gets larger. In mono-simulation case *Ref-1*, the chassis model needs to take a small integration step due to the stiff EPAS model. Instead, in co-simulation each solver can adapt to the local dynamics more efficiently.

In another mono-simulation case *Ref-2* with increased  $\Delta t_1$ , the CPU time reduces a lot as well but the time saving is not so effective as the co-simulation cases with a same or larger  $\Delta t_2$  setup. It can be observed that a big time saving is from a relaxation of communication with the chassis model.

The co-simulation case *CS-1* does not show an obvious advantage in the simulation speed. Because the adaptability of the local solver is constrained by a very frequent communication of 1 ms. In such a case the speed-up capability of co-simulation cannot be fully used even though the stiff part has been decoupled.

For other co-simulation cases, a further relaxation of  $\Delta t_2$  does increase the simulation speed but the improvement gets reduced at a larger step. If a rather large  $\Delta t_2$  has been taken, the two models can be seen as nearly decoupled and calculated independently. Therefore, the simulation time might just depend on the dynamics and solver of each part. In practice, the  $\Delta t_2$  size setup needs to be compromised considering the stability and coupling error which is discussed in the following section.



**Figure 8.** CPU time for a 5 seconds simulation.



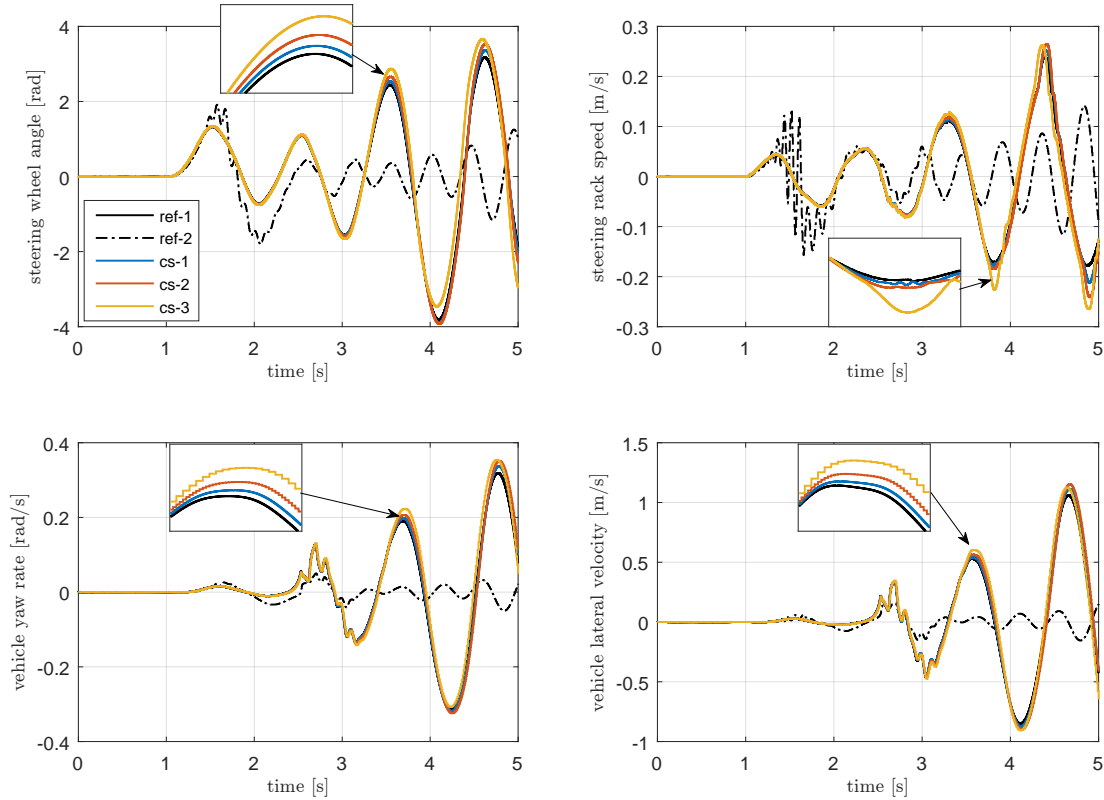


Figure 9. Simulation results of the EPAS system and chassis states.

## 5.2 Error analysis

Two EPAS system states (steering wheel angle  $\delta_s$ , rack speed  $\dot{x}_R$ ) and two chassis states (yaw rate, lateral velocity) from the previous simulation tests are plotted in Figure 9. It can be seen that *Ref-2* gives inaccurate and useless results although it can run really fast in the previous analysis. Because the high bandwidth coupling between the EPAS control, electric and the mechanism, the coupling variables are poorly approximated by extrapolation and the simulation error gets quite large.

The co-simulation cases, due to a more robust integration, have shown more stable and consistent results even their simulation speeds are faster. Case *CS-3* in Figure 9 shows larger stepwise signals from the chassis model. The co-simulation results deviate more at the peaks which is very intuitive since the accuracy of extrapolation is worse when the signal changes direction.

The relative global error  $\epsilon_{g,x}$  of selected state  $x$  are computed by the normalized root-mean-square error as:

$$\epsilon_{g,x} = \frac{\sqrt{\sum_{t=0}^T \left( (x_{cs}(t) - x_{ref}(t))^2 / T \right)}}{x_{ref}^{max} - x_{ref}^{min}} \quad (9)$$

where  $x_{ref}^{max}$  and  $x_{ref}^{min}$  are the maximum and minimum reference state value during simulation time  $t \in [0, T]$ . The relative global error  $\epsilon_{g,x}$  is plotted in Figure 10. One can

see that  $\epsilon_{g,x}$  in case *Ref-2* is clearly the worst and the error increases as the step  $\Delta t_2$  grows, which limits the relaxation of communication for the simulation speed-up. To reduce the error and enable a further relaxation, some explicit coupling methods from (Khaled et al., 2014) and (Benedikt et al., 2013) can be potentially applied, which is out of the scope of this paper. Thus, only a basic constant extrapolation is presented in this work.

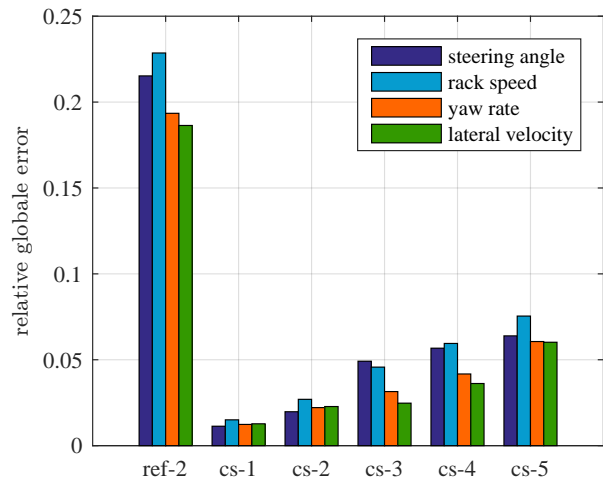
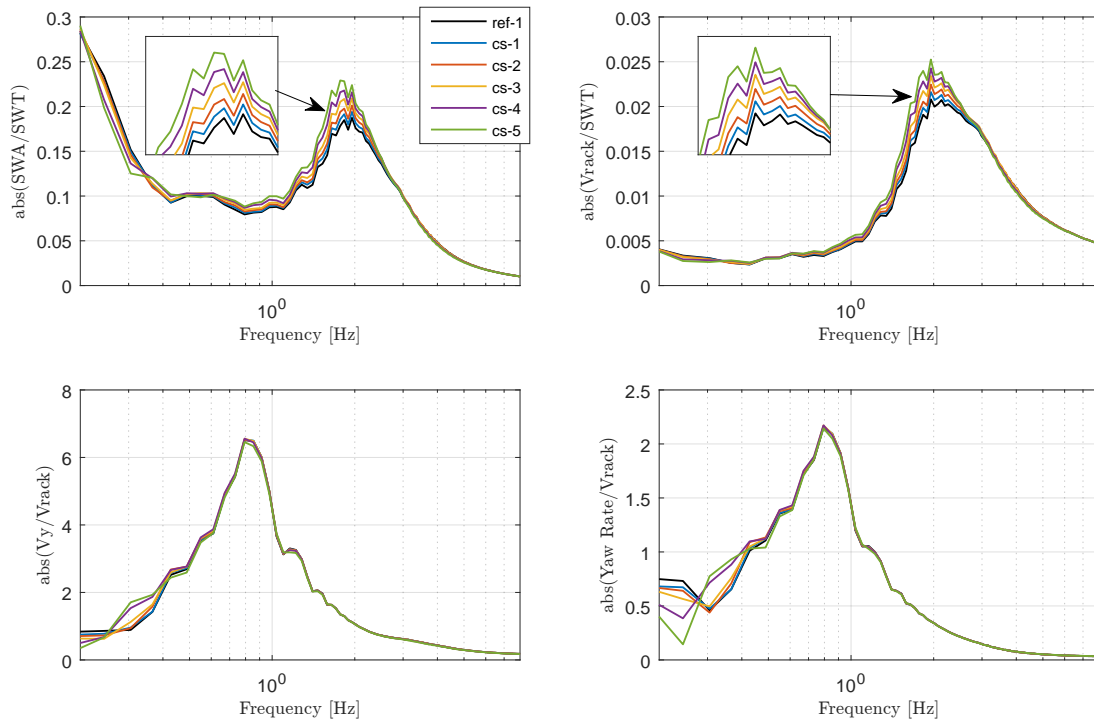


Figure 10. Comparison of the relative global error  $\epsilon_{g,x}$ .



**Figure 11.** The identified transfer behaviors of EPAS and chassis.

### 5.3 System dynamics analysis

To investigate the system dynamics in multiple conditions, a frequency domain comparison is more intuitive as a second analysis. In this analysis, a steering torque input from low frequency up to high frequency has been applied. The simulation time is long enough that the system can be excited sufficiently within the frequency range of interest. Two pairs of transfer functions are identified from the simulation results. The first is steering torque (SWT) to steering wheel angle (SWA) and rack speed (Vrack), which is more relevant to steering behavior. The second is from rack speed (Vrack) to chassis lateral speed (Vy) and yaw rate which mainly shows the chassis lateral dynamics.

The magnitudes of the transfer functions are plotted in Figure 11. The steering feedback character (SWA/SWT) and the EPAS transfer behavior (Vrack/SWT) are influenced in a certain range. The deviation gets larger around 1.1 Hz which is close to the chassis yaw eigenfrequency. As  $\Delta t_2$  increases, more delayed rack force resistant to the rack motion gives an increased steering wheel angle and rack speed.

The chassis transfer behaviors are relatively more consistent to the reference. It might be the reason that the chassis has a slower dynamics and more robust to the coupling effect. The deviation of chassis dynamics occurs mainly below 0.5 Hz and the magnitude of deviation is correlated to the relaxation condition.

Furthermore, the dynamics of the EPAS and chassis system limited the bandwidth of the coupling signals. In the high frequency range of steering input, the only exci-

tation to chassis system has been filtered out and the coupling effect gets minor.

## 6 Conclusion

In this paper a FMI-based co-simulation of EPAS and vehicle chassis system has been presented. The solver coupling approach is used for mechanical-functional system integration and also for mechanical systems in large time scale. The accelerated simulation speed makes the simulation tool more useful for design optimization and control tuning work. A controllable coupling error without severe numerical instability is induced by the explicit parallel calculation scheme. The approach can also be applied on real-time applications where the simulation speed is crucial. However, the CPU time from the current test is still huge that model order reduction might be needed to make each system real-time capable first.

The approach is quite promising for vehicle chassis and other mechatronic systems (e.g., active suspension, electric propulsion and automated driving system).

## Acknowledgement

The authors would like to thank ITEAM project funded by the European Union Horizon 2020 research and innovation program under Marie Skłodowska-Curie Grant Agreement No. 675999.

## References

Martin Aberger and Martin Otter. Modeling friction in Modelica with the Lund-Grenoble friction model. *Proceedings of the*

- 2nd International Modelica Conference*, 3:285–294, 2002.
- Martin Arnold. Stability of Sequential Modular Time Integration Methods for Coupled Multibody System Models. *Journal of Computational and Nonlinear Dynamics*, 5(3):031003, 2010.
- Karl Johan Astrom and Carlos Canudas de Wit. Revisiting the LuGre model; Stick-slip Motion and Rate Dependence. *IEEE Control Systems Magazine*, 6:101–114, 2008.
- Martin Benedikt, Daniel Watzenig, Josef Zehetner, and Anton Hofer. NEPCE - A nearly energy-preserving coupling element for weak-coupled problems and co-simulations. *V International Conference on Computational Methods for Coupled Problems in Science and Engineering*, pages 1–12, 2013.
- Martin Busch. *Zur effizienten Kopplung von Simulationsprogrammen*. PhD thesis, Kassel University, 2012.
- Weitao Chen, Shenhai Ran, and Bengt Jacobson. Design of Interface in Co-simulation for Electric Power Assisted Steering System Development. *Proceedings of the 14th International Symposium on Advanced Vehicle Control (AVEC' 18)*, 2018.
- Manfred Harrer and Peter Pfeffer. *Steering Handbook*. 2017.
- Abir Ben Khaled, Laurent Duval, Mohamed El Mongi Ben Gaïd, and Daniel Simon. Context-based polynomial extrapolation and slackened synchronization for fast multi-core simulation using fmi. In *International Modelica Conference*, pages 225–234. Linköping University Electronic Press, 2014.
- Salah Eddine Saidi, Nicolas Pernet, Yves Sorel, and Abir Ben Khaled. Acceleration of FMU Co-Simulation On Multi-core Architectures. *The First Japanese Modelica Conferences*, (124):106–112, 2016.



